AMENDMENTS TO CLAIMS

1. **(Currently Amended)** A ~~shared storage~~ distributed file system having a namespace defining a directory structure of files and metadata that includes pointers to real-data, the file system comprising:
   a) at least one server computer running server software for managing the namespace; and
   b) a plurality of client computers separate from the server computer, each running client software, the client software
      i) issuing namespace requests to the server computer, the namespace requests selected from the group consisting of requests to add new filenames to the namespace, requests to remove existing filenames from the namespace, and requests to search the namespace for filenames, and
      ii) directly retrieving, analyzing, and altering the metadata .

2. **(Original)** The file system of claim 1, wherein metadata includes allocation tables that store information identifying data as allocated and not allocated.

3. **(Original)** The file system of claim 1, wherein the client software directly generates metadata pointers to real-data.

4. **(Original)** The file system of claim 1, wherein the server software enforces file access permissions.

5. **(Currently Amended)** The file system of claim 1, wherein the server software manages the namespace in response to namespace requests from the client computers, including requests to ~~add new filenames to the namespace, to remove existing filenames from the namespace, and to search the namespace for filenames~~ read a directory from the namespace.

6. **(Original)** The file system of claim 5, wherein the namespace search for filenames returns information necessary to retrieve the metadata.

7. **(Original)** The file system of claim 6, wherein the server software enforces file permissions during the namespace search.

8. **(Original)** The file system of claim 1, wherein the client computers directly respond to file system requests concerning a file from an application program.

9. **(Original)** A network of connected computing devices for implementing a shared storage distributed file system, the file system having a namespace, real-data, and metadata, the network comprising:

   a) a network storage device connected to a network;

   b) a server computer that manages the namespace in response to namespace requests, including requests to add new filenames to the namespace and to remove existing filenames from the namespace; and

   c) a client computer in network communication with the server computer and the network storage device, wherein the client computer

      i) issues namespace requests to the server computer,

      ii) reads and writes the real-data directly from the network storage device, and

      iii) creates, retrieves, and modifies the metadata.

10. **(Original)** The network of claim 9, wherein the client computer communicates with the server computer via a local area network, and the client computer communicates with the network storage device via a storage area network.

11. **(Original)** The network of claim 10, wherein namespace requests are communicated via the local area network.

12. **(Original)** The network of claim 10, wherein the client reads and writes the real-data via the storage area network.

13. **(Original)** The network of claim 9, wherein the client computer requests file attributes from the server computer.

14. **(Original)** The network of claim 13, wherein file attributes are communicated via the local area network.

15. **(Original)** The network of claim 9, wherein the client computer acquires a lock prior to modifying the metadata.

16. **(Original)** The network of claim 9, wherein the namespace requests include requests to search the namespace for filenames.

17. **(Original)** The network of claim 16, wherein the server computer enforces file access permissions during the namespace search.

18. **(Original)** The network of claim 9, wherein the client computer stores the metadata by requesting that the metadata be stored by the server computer.

19. **(Original)** The network of claim 18, wherein the server computer stores the metadata within a real-data file.

20. **(Original)** The network of claim 18, wherein the server computer is operating as the server of a client-server file system to store the metadata.

21. **(Original)** The network of claim 18, wherein the server computer stores the metadata on a server storage device locally attached to the server computer.

22. **(Original)** The network of claim 18, wherein the server computer stores the metadata on a server storage device different from the network storage device.

23. **(Currently Amended)** An improved file system comprising:

    a) a storage device;

    b) a server software program that runs on a server computer and maintains a namespace based on requests from a client computer; and

    c) a client software program that runs on [a] the client computer that responds to file system requests from an application program concerning a file, wherein the client software

        i) obtains addressing metadata containing at least one pointer addressing real-data for the file,

        ii) uses the addressing metadata to locate real-data associated with the file on the storage device, and

        iii) alters the addressing metadata for the file.

24. **(Original)** The improved file system of claim 23, wherein the server software adds new filenames to the namespace, removes existing filenames from the namespace, and searches the namespace for filenames.

25. **(Original)** The improved file system of claim 24, wherein the client software sends a namespace search request to the server software in order to obtain addressing metadata.

26. **(Original)** The improved file system of claim 25, wherein the server software enforces file access permissions during the namespace search.

27. **(Original)** The improved file system of claim 25, wherein the addressing metadata is found within an inode obtained by the client software.

28. **(Original)** The improved file system of claim 23, wherein the client software further obtains allocation table metadata concerning allocation of storage on the storage device and modifies the allocation table metadata when performing file allocation and de-allocation.

29. **(Original)** The improved file system of claim 28, wherein the allocation table metadata is a bitmap table.

30. **(Original)** The improved file system of claim 23, wherein the addressing metadata is found within an inode obtained by the client software.

31. **(Original)** An improved file system comprising:

a) a storage device containing real-data and one or more direct pointers addressing the real-data;

b) a server software program that runs on a server computer, the server software program

i) maintains a namespace, and

ii) stores an indirect pointer within the namespace related to a file, the indirect pointer addressing at least one file related direct pointer on the storage device; and

c) a client software program that runs on a client computer that responds to file system requests from an application program concerning the file; the client software program

i) obtains the indirect pointer for the file from the server software program,

ii) uses the indirect pointer to obtain the file related direct pointer directly from the storage device, and

iii) uses the file related direct pointer to read and write real-data associated with the file directly from the storage device.

32. **(Original)** The improved file system of claim 31, wherein the client software program modifies the file related direct pointer during file write operations.

33. **(Original)** The improved file system of claim 32, wherein the client software acquires a lock prior to modifying the file related direct pointer.

34. **(Original)** The improved file system of claim 31, wherein the server software services namespace requests from the client computer, including requests to add new filenames to the namespace, to remove existing filenames from the namespace, and to search the namespace for filenames.

35. **(Original)** The improved file system of claim 31, wherein the client software further obtains allocation table metadata concerning allocation of storage on the storage device and updates the allocation metadata when performing file allocation and de-allocation.

36. **(Original)** The improved file system of claim 35, wherein the client software acquires a lock prior to updating the allocation table metadata.

37. **(Currently Amended)** An improved file system comprising:

a) a storage device;

b) a server computer running server software that maintains a namespace defining a directory structure of files, and

c) a client computer in network communication with the server computer and the storage device, the client computer running client software that

i) obtains allocation information concerning the allocation of storage on the storage device and

ii) alters the allocation information for regular files during file allocation and de-allocation;

wherein the alteration of the allocation information is performed in response to a request by an application program running on the client computer.

38. **(Original)** The improved file system of claim 37, wherein the server software accesses and modifies the directory structure in response to namespace requests from the client computer, including requests to add

new filenames to the namespace, to remove existing filenames from the namespace, and to search the namespace for filenames.

39. **(Original)** The improved file system of claim 37, wherein the client software acquires a lock prior to obtaining the allocation information.

40. **(Original)** The improved file system of claim 37, wherein the client software further:

    iii) obtains addressing metadata locating real-data for a particular file,

    iv) uses the addressing metadata to access real-data associated with the particular file on the storage device, and

    v) alters the addressing metadata for the file.

41. **(Original)** The improved file system of claim 40, wherein the client software acquires a lock prior to altering the addressing metadata.

42. **(Original)** The improved file system of claim 37, wherein the allocation information is obtained from the storage device and the altered allocation information is stored on the storage device.

43. **(Original)** A network of connected computer devices comprising:

    a) a first computer running software for

        i) managing a directory structure of files, and

        ii) servicing directory requests, the directory requests including requests to add filenames to the directory, remove filenames from the directory, and search the directory; and

    b) a second computer running software for

        i) submitting to the first computer directory requests relating to a file request, and

        ii) analyzing and altering metadata relating to the file request, the metadata including pointers to real-data.

44. **(Original)** The network of claim 43, wherein the metadata includes data allocation information.

45. **(Original)** The network of claim 43, wherein the second computer directly responds to file requests from an application program.

46. **(Original)** The network of claim 43, wherein the first computer enforces file access permissions for requests received from the second computer.

47. **(Original)** The network of claim 46, wherein the first computer enforces file access permissions while adding a filename to a directory.

48. **(Original)** The network of claim 46, wherein the first computer enforces file access permissions while removing a filename from a directory.

49. **(Original)** The network of claim 46, wherein the first computer enforces file access permissions while searching for a filename within a directory.

50. **(Currently Amended)** A method for handling a file request from an application, the file request relating to real-data of a file, the real-data being stored on a network connected~~, shared~~ storage device, the method comprising:

a) receiving the file request from the application at a client computer;

b) requesting an indirect extent pointer for the file from a server computer;

c) receiving the requested indirect extent pointer at the client computer;

d) using the indirect extent pointer to retrieve metadata from the storage device;

e) for a file read request, the client computer

i) analyzing the metadata to determine the locations of the real-data stored on the storage device, and

ii) reading the real-data from the storage device; and

f) for a file write request, the client computer

i) analyzing the metadata to determine the locations of the real-data stored on the storage device,

ii) allocating additional storage space to the file,

iii) writing real-data to the storage device,

iv) updating metadata, and

v) storing updated metadata on the storage device.

51. **(Original)** The method of claim 50, wherein during the allocation of additional storage space for the file write request, the client computer retrieves, analyses, and modifies the allocation table metadata.

52. **(Original)** A method for handling file requests by a file system comprising:

a) receiving the file request at a client computer;

b) requesting that a server computer perform a namespace search for the file of the request

c) analyzing and altering metadata relating to the location of real-data of the file request at the client computer; and

d) saving the metadata altered by the client computer.

53. **(Original)** The method of claim 52, wherein a lock is acquired prior to altering the metadata relating to the location of real-data.

54. **(Original)** A method for handling file requests by a file system comprising

a) receiving the file request at a client computer;

b) requesting namespace data relating to the file request from a server computer;

c) analyzing and altering metadata at the client computer, the metadata relating to the allocation information of storage on a storage device; and

d) saving the allocation information metadata altered by the client computer.

55. **(Original)** The method of claim 54, wherein the allocation information consists of bitmap tables.

56. **(Original)** The method of claim 54, wherein a lock is acquired prior to analyzing and altering allocation information metadata, and the lock is released after saving the allocation information.

57. **(New)** A computer product, comprising a computer readable medium having a computer program code embodied therein, said computer program code adapted to be executed to implement a method for handling file requests by a file system, the method comprising:

a) receiving the file request at a client computer;

b) requesting that a server computer perform a namespace search for the file of the request

c) analyzing and altering metadata relating to the location of real-data of the file request at the client computer; and

d) saving the metadata altered by the client computer.

58. **(New)** The file system of claim 1, wherein the file system is a shared storage distributed file system in which the client computers directly access the shared storage device.

59. **(New)** The network of claim 50, wherein the storage device is a shared storage device, whereby the client computer directly access the shared storage device.